09-07-00 A

Express Mail Label No. EK830777841US

UTILITY PATENT APPLICATION TRANSMITTAL

(Large Entity)

(Only for new nonprovisional applications under 37 C.F.R. 1.53(b))

Docket No.
POU9-2000-0048-US1

TO THE ASSISTANT COMMISSIONER FOR PATENTS
Box Patent Application
Washington, D.C. 20231

Transmitted herewith for filing under 35 U.S.C. 111(a) and 37 C.R.F. 1.53(b) is a new utility patent application for an invention entitled:

High Speed Software Driven Emulator Comprised of a Plurality of Emulation Processors With A Method To Allow Memory Read/Writes Without Interrupting the Emulation

and invented by:

William F. BEAUSOLEIL, R. Bryan COOK, Tak-kwong NG, Helmut ROTH, Peter TANNENBAUM, Lawrence A. THOMAS, and Norton J. TOMASSETTI

If a CONTINUATION APPLICATION, *check appropriate box and supply the requisite information*:
☐ **Continuation** ☐ **Divisional** ☐ **Continuation-in-part (CIP)** of prior application No.: _____
Which is a:
☐ **Continuation** ☐ **Divisional** ☐ **Continuation-in-part (CIP)** of prior application No.: _____
Which is a:
☐ **Continuation** ☐ **Divisional** ☐ **Continuation-in-part (CIP)** of prior application No.: _____

Enclosed are:

Application Elements

1. ☒ Filing fee as calculated and transmitted as described below
2. ☒ Specification having 12 pages and including the following:
   a. ☒ Descriptive Title of the Invention
   b. ☒ Cross References to Related Applications (*if applicable*)
   c. ☐ Statement Regarding Federally-sponsored Research/Development (*if applicable*)
   d. ☐ Reference to Microfiche Appendix (*if applicable*)
   e. ☒ Background of the Invention
   f. ☒ Brief Summary of the Invention
   g. ☒ Brief Description of the Drawings (*if drawings filed*)
   h. ☒ Detailed Description
   i. ☒ Claim(s) as Classified Below
   j. ☒ Abstract of the Disclosure

UTILITY PATENT APPLICATION TRANSMITTAL (Large Entity) (Only for new nonprovisional applications under 37 C.F.R. 1.53(b))	Docket No. POU9-2000-0048-US1

Application Elements (Continued)

3.☒ Drawing(s) (*when necessary as prescribed by 35 USC 113*)

a. ☐ Formal Number of Sheets _____

b. ☒ Informal Number of Sheets 2

4. ☒ Oath or Declaration

a. ☒ Newly executed (*original or copy*) ☐ Unexecuted

b. ☐ Copy from a prior application (37 CRF 1.63(d)) (*for continuation/divisional application only*)

c. ☐ With Power of Attorney ☐ Without Power of Attorney

d. ☐ DELETION OF INVENTOR(S)
Signed statement attached deleting inventor(s) named in the prior application, see 37 C.F.R. 1.63(d)(2) and 1.33(b).

5. ☐ Incorporation By Reference (*usable if Box 4b is checked*)
The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated herein by reference therein.

6. ☐ Computer Program in Microfiche (*Appendix*)

7. ☐ Nucleotide and/or Amino Acid Sequence Submission (*if applicable, all must be included*)

a. ☐ Paper Copy

b. ☐ Computer Readable Copy (*identical to computer copy*)

c. ☐ Statement Verifying Identical Paper and Computer Readable Copy

Accompanying Application Parts

8. ☒ Assignment Papers (*cover sheet & document(s)*)

9. ☐ 37 CFR 3.73(B) Statement (*when there is an assignee*)

10. ☐ English Translation Document (*if applicable*)

11. ☒ Information Disclosure Statement/PTO-1449 ☒ Copies of IDS Citations

12. ☐ Preliminary Amendment

13. ☒ Acknowledgment Postcard

14. ☒ Certificate of Mailing

☐ First Class ☒ Express Mail (*Specify Label No.*): EK830777841US

UTILITY PATENT APPLICATION TRANSMITTAL (Large Entity) (Only for new nonprovisional applications under 37 C.F.R. 1.53(b))	Docket No. POU9-2000-0048-US1

Accompanying Application Parts (Continued)

15. ☐ Certified Copy of Priority Document(s) (*if foreign priority is claimed*)

16. ☐ Additional Enclosures (*please identify below*):

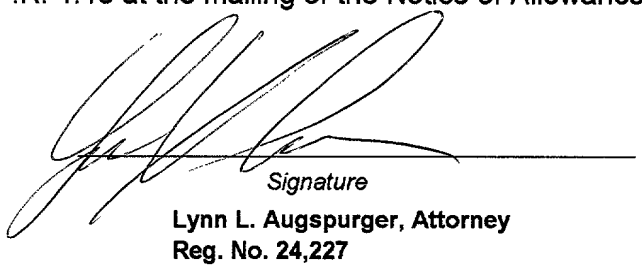
Fee Calculation and Transmittal

CLAIMS AS FILED					
FOR	**#FILED**	**#ALLOWED**	**#EXTRA**	**RATE**	**FEE**
Total Claims	4	- 20 =	0	x $18.00	$0.00
Indep. Claims	1	- 3 =	0	x $ 78.00	$0.00
Multiple Dependent Claims (check if applicable) ☐					$0.00
				BASIC FEE	$690.00
OTHER FEE (*specify purpose*) _____					
				TOTAL FILING FEE	$690.00

☐ A check in the amount of to cover the filing fee is enclosed.

☒ The Commissioner is hereby authorized to charge and credit Deposit Account No. **09-0463** as described below. A duplicate copy of this sheet is enclosed.

- ☒ Charge the amount of $690.00 as filing fee.
- ☒ Credit any overpayment.
- ☒ Charge any additional filing fees required under 37 C.F.R. 1.16 and 1.17.
- ☐ Charge the issue fee set in 37 C.F.R. 1.18 at the mailing of the Notice of Allowance, pursuant to 37 C.F.R. 1.31(b).



Signature

Dated: September 6, 2000

Lynn L. Augspurger, Attorney
Reg. No. 24,227
IBM Corporation
Intellectual Property Law
2455 South Rd., M/S P386
Poughkeepsie, NY 12601
Phone: (845) 433-1174
Fax: (845) 432-9786

Docket Number: POU9-2000-0048-US1

HIGH SPEED SOFTWARE DRIVEN EMULATOR COMPRISED OF A PLURALITY OF EMULATION PROCESSORS WITH A METHOD TO ALLOW MEMORY READ/WRITES WITHOUT INTERRUPTING THE EMULATION

APPLICATION FOR

UNITED STATES LETTERS PATENT

"Express Mail" Mailing Label No.: EK830777841US

Date of Deposit: September 6, 2000

I hereby certify that this paper is being deposited with the United States Postal Service as "Express Mail Post Office to Addressee" service under 37 CRF 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, D.C. 20231.

Name: Billy R. Stacy

Signature: Billy R. Stacy
Billy R. Stacy

INTERNATIONAL BUSINESS MACHINES CORPORATION

Title: HIGH SPEED SOFTWARE DRIVEN EMULATOR COMPRISED OF A PLURALITY OF EMULATION PROCESSORS WITH A METHOD TO ALLOW MEMORY READ/WRITES WITHOUT INTERRUPTING THE EMULATION

Cross Reference to Related Applications:

The following copending applications, assigned to the assignee of the present invention, contain common disclosure and are incorporated herein by reference in their entireties:

"High Speed Software Driven Emulator Comprised of a Plurality of Emulation Processors with Improved Board-to-Board Interconnection Cable Length Identification System," Serial No. _____, filed _____, (Attorney Docket No. POU9-2000-0045-US1).

"High Speed Software Driven Emulator Comprised of a Plurality of Emulation Processors with an Improved Maintenance Bus that Streams Data at High Speed," Serial No. _____, filed _____, (Attorney Docket No. POU9-2000-0046-US1).

"High Speed Software Driven Emulator Comprised of a Plurality of Emulation Processors with a Method to Allow High Speed Bulk Read/Write Operation Synchronous DRAM While Refreshing the Memory," Serial No. _____, filed _____, (Attorney Docket No. POU9-2000-0047-US1).

"High Speed Software Driven Emulator Comprised of a Plurality of Emulation Processors with Improved Multiplexed Data Memory," Serial No. _____, filed _____, (Attorney Docket No. POU9-1999-0183-US1)

Field of the Invention:

This invention relates to software driven emulators in which a large number of processors operate in parallel to emulate complex logical functions, and more particularly, to a system and method for managing the memory arrays of such emulators so that memory data transfers can be executed without interrupting the emulation process.

Trademarks:

Background:

The usefulness of software driven emulators has increased enormously with growth in the complexity of integrated circuits. Basically, an emulation engine operates to mimic the logical design of a set of one or more integrated circuit chips. The emulation of these chips in terms of their logical design is highly desirable for several reasons. The utilization of emulation engines has also grown up with and around the corresponding utilization of design automation tools for the construction and design of integrated circuit chip devices. In particular, as part of the input for the design automation process, logic descriptions of the desired circuit chip functions are provided. The existence of such software tools for

processing these descriptions in the design process is well suited to the utilization of emulation engines which are electrically configured to duplicate the same logic function that is provided by a design automation tool.

Utilization of emulation devices permits testing and verification via electrical circuits of logic designs before these designs are committed to a socalled "silicon foundry" for manufacture. The input to such foundries is the functional logic description required for the chip and its output is initially a set of photolithographic masks which are then used in the manufacture of the desired electrical circuit chip device. Verifying that logic designs are correct in the early stage of chip manufacturing eliminates the need for costly and time-consuming second passes through a silicon foundry.

Another advantage of emulation systems is that they provide a device that makes possible the early validation of software meant to operate the emulated chips. Thus, software can be designed, evaluated and tested well before the time when actual circuit chips become available. Additionally, emulation systems can also operate as simulator-accelerator devices thus providing a high-speed simulation platform.

Emulation engines of the type contemplated by this invention contain an interconnected array of emulation processors (EP). Each emulation processor (hereinafter also sometimes simply referred to as "processor") can be programmed to evaluate logic functions (for example, AND, OR, XOR, NOT, NOR, NAND, etc.). The program driven processors operate together as an interconnected unit, emulate the entire desired logic design. However, as integrated circuit designs grow in size, more emulation processors are required to accomplish the emulation task. An aim,

therefore, is to increase the capacity of emulation engines in order to meet the increasingly difficult task of emulating more and more complex circuits and logic functions by increasing the number of emulation processors in each of its modules.

For purposes of better understanding the structure and operation of emulation devices generally, and this invention particularly, United States Patent No. 5,551,013 and patent application Serial No. 09/373,125 filed August 12, 1999, both of which are assigned to the assignee of this application, are hereby incorporated herein by reference.

Patent No. 5,551,013 shows an emulation chip, called a module here, having multiple (e.g. 64) processors. All processors within the module are identical. The sequencer and the interconnection network occur only once in a module. The control stores hold a program created by an emulation compiler for a specified processor. The stacks hold data and inputs previously generated and are addressed by fields in a corresponding control word to locate the bits for input to the logic element. During each step of the sequencer an emulation processor emulates a logic function according to the emulation program. The data flow control interprets the current control word to route and latch data within the processor. The node-bit-out signal from a specified processor is presented to the interconnection network where it is distributed to each of the multiplexers (one for each processor) of the module. The node address field in the control word allows a specified processor to select for its node-bit-in signal the node-bit-out signal from any of the processors within its module. The node bit is stored in the input stack on every step. During any operation, the node-bit-out signal of a specified processor may be accessed by none, one, or all of the processors within the module.

Data routing within each processor's data flow and through the interconnection network occurs independently of and overlaps the execution of the logic emulation function in each processor. Each control store stores control words executed sequentially under control of the sequencer and program steps in the associated module. Each revolution of the sequencer causes the step value to advance from zero to a predetermined maximum value and corresponds to one target clock cycle for the emulated design. A control word in the control store is simultaneously selected during each step of the sequencer. A logic function operation is defined by each control word.

Each of these emulation processors has an execution unit for processing multiple types of logic gate functions. Each emulation processor switches from a specified one logic gate function to a next logic gate function in a switched-emulation sequence of different gate functions. The switched-emulation sequence of each of the processors thus can emulate a subset of gates in a hardware arrangement in which gates are of any type that the emulation processors functionally represent for a sequence of clock cycles. The processors are coupled by a like number of multiplexers having outputs respectively connected to the emulation processors of a module and having inputs respectively connected to each of the other emulation processors. The bus connected to the multiplexers enables an output from any emulation processor to be transferred to an input of any other of the emulation processors. In accordance with the teachings of the pending application, the basic design of the 5,551,013 patent is improved by interconnecting processors into clusters. With the processors cascaded, all processors in a cluster perform the setup and storing of results in parallel. This setup includes routing of the data through multiple evaluation units for the

evaluation phase. For most efficient operation, the input stack and data stack of each processor must be stored in shared memory within each cluster. Then, all processors perform the storage phase, again in parallel. The net result is multiple cascaded evaluations performed in a single emulation step. Every processor in a cluster can access the input and data stacks of every other processor in the cluster and less space on each module chip for the functions that support the processor operation, particularly the memory functions.

The ET4 emulator can be partitioned so that the emulator is capable of running more than one emulation at a time. To fully exploit this capability, there is a need to read from and write to the "main data memories". Each module has a main data memory, comprised of two commercially available Sync DRAMs and one commercially available SSRAM. Each emulator board is comprised of sixty-five modules and a memory controller serves all modules on a board. In prior art emulators of the type contemplated by this invention, an emulation process is stopped in order to read or write data to a main module memory from a source external to the emulation process.

Summary of the Invention:

An object of this invention is the provision of a method that enables transfers of external data to and from a module's main data memories without interrupting emulation in progress.

Briefly, this invention contemplates the provision of a software driven emulator in which the stored emulation program for a processor module is compiled to include a code bit or bits in the emulation instruction step sequence that is decoded as main data memory disable command. Thus, once in each emulation program

cycle, the memory controller disables the main data memories on the module, and allows the maintenance bus to read or write data to these memories.

Brief Description of the Drawings:

The foregoing and other features and advantages of the invention will be described with reference to the accompanying drawings. In the drawings, like reference numbers generally indicate identical, functionally similar, and/or structurally similar elements. Also in the figures, the left most digit of each reference number corresponds to the figure in which the reference number is first used.

Figure 1 is a high level block diagram of two of the modules on a printed circuit board of the type to which this invention is applicable.

Figure 2 is a logic block diagram illustrating the system for reading and/or writing to the main module memories without interrupting the emulation process in accordance with the teachings of this invention.

Detailed Description of the Invention:

Referring now to Figure 1, an emulator of the type contemplated by this invention is comprised of a number of printed circuit boards 14 interconnected by a maintenance bus, which also connects each board to a work station 26. Work station 26 serves, among other functions, to load data to the emulator memories, and receive data from the emulator memories. Each printed circuit board has a memory controller 22, which provides a memory control interface function for the modules on a printed

circuit board. Each module (only two are illustrated in Figure 1) includes a processor chip 16 (ET4) and three RAM chips 18 (two SDRAMs and one SSRAM), which are coupled to the ET4 processor 16. Each ET4 chip is comprised of sixty-four one-bit processors.

Referring now to Figure 2 as well the preceding figure, each of the 64 processors in the module includes a control store 30, which in this illustrative embodiment is 256 steps deep with 96 bits wide processor control codes that determine the logic function of the associated processor 31 during that step in the emulation process. As will be appreciated by those skilled in the art, in carrying out an emulation, a compiler program compiles the logic design as a series of processor instruction steps that are stored in the control store 30 of each processor in a module. In accordance with the teaching of this invention, in order to allow reading and/or writing to the RAM memories of a module, the control store program in one of the control store memories is compiled so that it has a bit in a bit location that is decoded as command to memory controller 22 to disable all module processors from writing to or reading from the module RAM memories 18. As illustrated in Figure 2, bit "M" in step "N" is set to a predetermined binary value ("0" or "1"), decoder 33 decodes the state of bit "M" in step "N" and sends a disable command to memory controller over the maintenance bus. This allows work station 22 to transfer data to or from the module's RAM memories during this step in each cycle of control store 30. When the program progresses to the next step, the disable command terminates and the processors are again enabled for data transfer to and from the RAMs. Of course, the program could be compiled and decoded to provide two or more interrupts each cycle of the control store.

While the preferred embodiment to the invention has been described, it will be understood that those skilled in the art, both now and in the future, may make various improvements and enhancements which fall within the scope of the claims which follow. These claims should be construed to maintain the proper protection for the invention first described.

What is claimed is:

Claim 1. In an emulation engine comprised of a plurality of modules, a work station, and a maintenance bus for transferring data between the work station and said modules, each of said modules including a plurality of module processors and a module main memory unit accessible for data transfers during an emulation by each of said plurality of processors, each of said processors having a control store to store a programmable sequence of emulation steps that define logic states for its processor, a method to allow data transfers between said module main memory unit and said work station without interrupting an in progress emulation, including the steps of:

compiling said programmable sequence of emulation steps to include, in at least one step, a blocking code that is decoded, when the step is read from the control store, as a disable command between the plurality of module processors and said module main memory;

decoding said blocking code and, in response thereto, blocking transfers between the plurality of module processors and said module main memory; and

transferring data between said work station and said module main memory while transfers between the plurality of module processors and said module main memory are blocked.

Claim 2. A method to allow data transfers between said module main memory unit and said work station as in claim 1 further including the step of unblocking transfers between the plurality of module processors and said module main memory when the step is decoded that is next in the sequence after said step that includes said blocking code.

Claim 3. A method to allow data transfers between said module main memory unit and said work station as in claim 1 wherein said programmable sequence is repeated and said decoding and transferring steps are repeated with each repetition of said programmable sequence.

Claim 4. A method to allow data transfers between said module main memory unit and said work station as in claim 2 wherein said programmable sequence is repeated and said decoding and transferring steps are repeated with each repetition of said programmable sequence.

Title: HIGH SPEED SOFTWARE DRIVEN EMULATOR COMPRISED OF A PLURALITY OF EMULATION PROCESSORS WITH A METHOD TO ALLOW MEMORY READ/WRITES WITHOUT INTERRUPTING THE EMULATION

Abstract of the Disclosure:

A software driven emulator in which the stored emulation program for a processor module is compiled to include a code bit or bits in the emulation instruction step sequence that is decoded as main data memory disable command. Thus, once in each emulation program cycle, the memory controller disables the main data memories on the module, and allows the maintenance bus to read or write data to these memories.